AMENDMENTS TO THE CLAIMS

The following is a complete, marked-up listing of revised claims with a status identifier in parenthesis, underlined text indicating insertions, and strike through and/or double-bracketed text indicating deletions. The following listing of claims will replace all prior versions and listings of claims in the application.

LISTING OF CLAIMS

1.   (Currently Amended)   A method for coding a structured document, the method comprising:

generating a plurality of codes using at least one name space and allocating the plurality of codes for data types defined by the at least one name space;

assigning the at least one name space to another name space, such that at least one assignment information is generated, the assignment information including at least one inheritance relationship between the bequeathing namespace and the inheriting namespace,

wherein the assignment information of the inheriting namespace includes at least one code of a basic data type of the bequeathing namespace for a header type of the inheriting namespace,

the basic data type of the bequeathing namespace being a data type from which the header type of the inheriting namespace originates directly or the basic data type of the bequeathing name space being a basic data type of a yet another bequeathing name space in an inheritance hierarchy, the inheritance hierarchy including the yet another bequeathing name space, the bequeathing name space and the inheriting name space; and

allocating respectively separate codes, independent of other schemas or name spaces and for elements either defined or declared in the schemas, name

spaces, or in groups of schemas or name spaces, for another schema, a name space or for another group of schemas and name spaces, the allocation including:

sorting all data types of a name space, which are bequeathed from data types of other name spaces, in a list in a sequence of global type codes of respective basic data types as defined in the MPEG-7 standard, the respective basic data types being the data types in other name spaces, from which the sorted data types are bequeathed;

lexicographically sorting in each instance data types of a name space, which are bequeathed from a specific basic data type of a specific other name space;

sorting according to the sequence defined in the MPEG-7 standard into the existing list of data types all the data types of a name space, which are not bequeathed from a data type of another name space; and

allocating the separate codes in list sequence to the data types of the name space.

2.   (Previously Presented)   The method as claimed in claim 1, wherein a subset comprising addressable data types of the inheriting name space is determined based on an initial basic type of the bequeathing name space and on the basis of the inheritance relationship between the bequeathing and inheriting name spaces.

3.   (Previously Presented)   The method as claimed in claim 2, wherein the addressable subset is determined based on the initial basic type by establishing the basic types of the bequeathing name space.

4.   (Previously Presented)   The method as claimed in claim 3, wherein, based on the initial basic type of the bequeathing name space, the header type is determined in the inheriting name space and the basic type of the inheriting name space is identified from the bequeathing name space using the assignment information, the initial basic type of the inheriting name space being a basic type of the bequeathing name space.

5.   (Previously Presented)   The method as claimed in claim 1, wherein the assignment information assigned to the inheriting name spaces is stored together with the respective name space in a first device carrying out at least one of the coding and a decoding operation.

6.   (Previously Presented)   The method as claimed in claim 5, wherein the assignment information assigned to the inheriting name spaces is generated in a second device and transmitted together with the respective name space to the first device carrying out at least one of the coding and decoding operation.

7.   (Cancelled)

8.   (Currently Amended)   The method as claimed in claim 1~~claim 7~~, wherein, to identify the at least one of schema, name space and the group of at least one of schemas and name spaces, the respectively separate codes are subdivided into corresponding address areas.

9.   (Currently Amended)   The method as claimed in claim 1~~claim 7~~, wherein the separate codes respectively comprise a local code at least one of relating to at least one of the schema and the name space and relating to at least one of the

group of schemas and name spaces and an identification code to identify at least one of the schema, name space and the group of at least one of schemas and name spaces.

10.   (Currently Amended)   The method as claimed in <u>claim 1</u>claim 7, wherein separate codes are generated for at least one of global elements, substitution groups and data types.

11.   (Previously Presented)   The method as claimed in claim 10, wherein separate codes are generated for data types type codes such that within the inheritance tree of a name space, the data type adjacent to a first data type in the same name space is at a code interval with respect to the first data type, said code interval corresponding to the number of data types derived from the first data type in this name space.

12.   (Cancelled)

13.   (Previously Presented)   A method for decoding a structured document, comprising:

        decoding a document previously coded according to a method as claimed in claim 1.

14.   (Previously Presented)   A method for decoding a structured document, comprising:

        decoding a document previously coded according to a method as claimed in claim 11 wherein, to decode a binary type code, the code length of the separate

codes of the binary type codes is determined from the number of derived data types.

15.   (Previously Presented)   A method for decoding a structured document, comprising:

decoding a document previously coded according to a method as claimed in claim 4, wherein, to decode a specific type code, a sub-tree of the inheritance tree of the name space in which the specific type code is located is determined from the code intervals between adjacent data types.

16.   (Previously Presented)   A method for decoding an XML-based document, comprising:

decoding a document previously coded according to a method as claimed in claim 1.

17.   (Previously Presented)   A method for decoding an XML-based document, comprising:

decoding a document previously coded according to a method as claimed in claim 11, wherein, to decode a binary type code, the code length of the separate codes of the binary type codes is determined from the number of derived data types.

18.   (Previously Presented)   A method for decoding an XML-based document, comprising:

decoding a document previously coded according to a method as claimed in claim 11, wherein, to decode a specific type code, a sub-tree of the inheritance tree

of the name space in which the specific type code is located is determined from the code intervals between adjacent data types.

19. (Previously Presented) A method for decoding an XML-based document, comprising:

decoding a document previously coded according to a method as claimed in claim 11, wherein, to determine the basic types originate from an initial basic type, code length is determined from the code intervals between adjacent data types.

20. (Previously Presented) A method for decoding an XML-based document, comprising:

decoding a document previously coded according to a method as claimed in claim 11, wherein, to determine the number of types in the subset, code length is determined based on the header types from the code intervals between adjacent header types.

21-24. (Cancelled)

25. (Previously Presented) A method as claimed in claim 1, for coding an XML-based document.

26. (Previously Presented) A method as claimed in claim 13, for decoding an XML-based document.

27. (Cancelled)

*** END CLAIM LISTING ***